① NW LEVEL II

AD A074854

COMPUTER SCIENCE TECHNICAL REPORT SERIES

D D C
RECEIVED
OCT 11 1979
A

UNIVERSITY OF MARYLAND
COLLEGE PARK, MARYLAND
20742

DDC FILE COPY

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

79 10 09 125

15 DAAG53-76-C-Ø138,
DARPA Order-32Ø6

9 Technical rept.,

14 TR-689
DAAG53-76C-0138

11 Sep~~tember~~ 1978

Accession For
NTIS GRA&I ☑
DDC TAB □
Unannounced □
Justification ______

By ______

Distribution/

Availability Codes

Dist. A | Avail and/or special

6 A COMPARISON OF NOISE CLEANING TECHNIQUES.

10 Judith P. / Davenport
Defense Mapping Agency
Building 56
U.S. Naval Observatory
Washington, D.C. 20305

12 40

ABSTRACT

A variety of noise cleaning techniques are compared on two test pictures. All of the techniques tested are based on local analysis of a small neighborhood of each pixel. They include median and mode filtering; selective local averaging or weighted averaging schemes of various sorts; and Kalman filtering. Iterated weighted averaging and iterated median filtering were judged to give the best overall performance.

The support of the Defense Advanced Research Projects Agency and the U.S. Army Night Vision Laboratory under Contract DAAG53-76C-0138 (DARPA Order 3206) is gratefully acknowledged, as is the help of Ms. Kathryn Riley in preparing this paper. During the period of this work the author was on a full time training assignment from the Defense Mapping Agency Aerospace Center.

JOB

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

403 018

1. Introduction

Many types of noise can be present in a picture [1]. Each type interacts with the original image in a different way. For example, additive noise, such as channel noise, may be uncorrelated from point to point and independent of the picture signal. The picture can then be represented by $g = f+\nu$, where f is the input picture and ν is the noise. Uncorrelated, multiplicative noise, such as the noise present in a picture scanned by a flying spot scanner, is another type. If this noise is proportional to the signal, it can be expressed as $g = f+\nu_1 f=(1+\nu_1)f = f\nu$.

Film grain noise is an example of noise which is dependent on the gray levels of the point in question and those of nearby points. Both additive and multiplicative noise may be present.

The noise ν itself can be statistically characterized in various ways. A common assumption is that it is Gaussian, i.e. that its probability density has the form

$$\frac{1}{\sqrt{2\pi}\sigma} e^{-(\nu-\mu)^2/2\sigma^2}$$

where μ is the mean noise value (often assumed to be zero) and σ is the noise standard deviation. This paper deals primarily with additive Gaussian noise.

Many different noise cleaning methods have been proposed. Most of them operate in the space domain, and are based on comparison of the gray level of each pixel with the gray levels of the pixels in its neighborhoods. These comparisons can usually be carried out in parallel (i.e., independently) for all pixels, though they may also be iterated [2,3]. A variety of such methods are implemented and compared in this paper. For comparison, a classical sequential method, Kalman filtering, was also implemented. It this method, the computations for the individual pixels are not independent, but are done in a fixed sequence, so that the gray level correction for a given pixel depends on those already computed for previously examined pixels.

The methods were applied to the two noisy pictures in Figure 1. Figure 1a is a 128x128 image of an octagon of gray level 33 (on a 0-63 scale) on a background of gray level 28. Figure 1c is a 127x127 image of a section of a LANDSAT IR band of California (displayed as a negative). Gaussian noise was added to Figures 1a and 1c to obtain Figures 1b ($\mu=0$, $\sigma=5$) and 1d ($\mu=0$, $\sigma=8$).

2. Methods and results

With the exception of the Kalman filter, all methods implemented operated on a 3x3 neighborhood centered at a point P, unless otherwise indicated. The methods in Sections 2.1-2.7 were iterated; those in Sections 2.8-2.9 were not. In defining the methods, the point P is assumed to be at (x,y), so that its neighborhood is

(x-1,y+1)	(x,y+1)	(x+1,y+1)
(x-1,y)	(x,y)	(x+1,y)
(x-1,y-1)	(x,y-1)	(x+1,y-1)

The gray level at (u,v) is denoted by f(u,v). The noisy image is represented by $g(m,n) = f(m,n) + \nu(m,n)$, where $\nu(m,n)$ is the additive Gaussian noise.

2.1. Mode filtering [4]

In this method of noise cleaning, the gray level at a point P is replaced by the mode (most frequently occurring value) of the gray levels in an nxn neighborhood of P.

Since the mean noise value is zero, the expected value of the noisy picture is the same as that of the original picture:

$$\begin{aligned} E\{g(i,j)\} &= E\{f(i,j)+\nu(i,j)\} \\ &= E\{f(i,j)\} + E\{\nu(i,j)\} \\ &= E\{f(i,j)\} \end{aligned}$$

Hence, if the original picture is approximately constant in the neighborhood of P, the most common value occurring in the noisy picture should be that constant value. Thus, one can expect that mode filtering will remove the noise present in the test pictures. However, with this method, edge points (where the original gray level is not approximately constant) will often be misclassified, causing edges to be blurred.

The results of four iterations of mode filtering are shown in Figure 2. Both images have a blotchy appearance and edges are not preserved. In this case, the image worsened with each iteration.

2.2. Median filtering [5,6]

This method replaces the gray level at a point P by the median of the gray levels in an nxn neighborhood of P. Median filtering, like mode filtering, is good for smoothing out noise because it rejects extreme values. It also does not blur sharp edges. This can be shown by considering a one dimensional example. Suppose the given sequence of values is

...,0,0,0,0,1,0,0,0,...

Applying mean filtering over a 1x3 neighborhood yields

$\ldots,0,0,\frac{1}{3},\frac{1}{3},\frac{1}{3},0,\ldots$

whereas applying median filtering gives

...,0,0,0,0,0,0,...

Similarly, for the input sequence

...,0,0,0,0,1,1,1,1,..

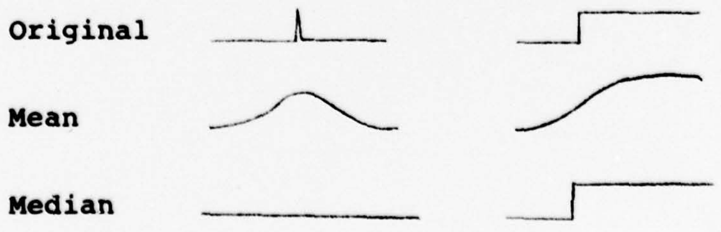
mean filtering gives

$\ldots,0,0,\frac{1}{3},\frac{2}{3},1,1,\ldots$

while median filtering gives

...,0,0,0,1,1,1,...

These effects are shown schematically in the following diagram:



Four iterations of this method are shown in Figure 3. The first iteration of the octagon seems to be the most pleasing, although it is mottled. The image becomes smoother with each successive iteration, but the edges seem somewhat more blurred. By contrast, the LANDSAT image improves with each iteration.

2.3. E^k [3]

In this class of methods, the gray level at a point P is replaced by the average gray level of those k neighbors (in an nxn neighborhood of P) whose gray levels are closest to that of P. If the non-noisy gray levels are $z_1,\ldots,z_k$, and the noise levels at these points are $w_1,\ldots,w_k$, then the average $[(z_1+w_1)+\ldots+(z_k+w_k)]/k = [z_1+\ldots+z_k]/k + [w_1+\ldots+w_k]/k$, where $[w_1+\ldots+w_k]/k$ can be regarded as a sample of a random variable with mean 0 and standard deviation $^{\sigma}/\sqrt{k}$ [1]. This averaging reduces the amplitude of the noise fluctuations. Thus, the image should become smoother, although possibly more blurred, with each iteration.

Values of k were chosen to be 2,4,6,8. E^8 is just an averaging of all the points surrounding P. E^2 should preserve lines and edges the best as, in theory, the two points along the same line or edge as P should have gray levels closest to that of P. E^4 should preserve edges fairly well; for a straight edge, five of P's neighbors should be on the same side of the edge as P. E^6 and E^8 will both blur edges. In addition, they will cause sharp protuberances in a region to disappear.

Four iterations for each value of k are shown in Figure 4. The results are similar to those found in [3]. E^2 preserved edges but did little noise cleaning. E^4 removed more noise than E^2, but less than E^6. Edges are somewhat more jagged for E^4, but E^6 begins to blur the image after two iterations.

2.4. Gradient smoothing [1,3]

The gray level at a point P is replaced by the average of the neighbors (in an nxn neighborhood of P) which have lower gradient values. The Roberts approximation to the digital gradient was implemented. It is computed using $\max(|g(i,j)-g(i+1,j+1)|, |g(i+1,j)-g(i,j+1)|)$.

The gradient serves as a measure of the differences in gray levels between neighboring points. In regions where the gray levels are similar, the gradient values will all be small, so that gradient smoothing will replace the gray level at P with the average gray level of a randomly chosen set of neighbors. On the other hand, at an edge, gradient values will be lower for neighbors that are farther away from the edge, i.e. that are interior to the region to which P belongs, so that P will usually be replaced by the average of these neighbors.

Results of four iterations of gradient smoothing are shown in Figure 5. Iteration produced smoother pictures, but somewhat blurred the edges.

2.5. Selective Averaging [1]

Three variations of this type of method were implemented. One variation consists of replacing the gray level at a point P by the average of its eight neighbors only if it differs from at least six of its neighbors by a given threshold value. Threshold values of $t = 2,3$ were used for the octagon where $\sigma = 5$ and $t = 3,4,5$ for the LANDSAT image where $\sigma = 8$. These values were chosen to be roughly the same fractions of the respective standard deviations. The results are shown in Figure 6a. As expected, smaller t's yield greater smoothing, and in all cases the images become somewhat blurred or ragged.

A second variation computes the absolute differences d_1 and d_2 where

$$d_1 = |[(g(i-1,j+1)+g(i,j+1)+g(i+1,j+1)] - [g(i-1,j-1)+g(i,j-1)+g(i+1,j-1)]|/3$$

and

$$d_2 = |[g(i-1,j+1)+g(i-1,j)+g(i-1,j-1)] - [g(i+1,j+1)+g(i+1,j)+g(i+1,j-1)]|/3$$

If the larger of these absolute differences is greater than a given threshold t, $d_m = \max(d_1,d_2) > t$, the point P is replaced by the average of the two neighbors in the direction $\tan^{-1}(d_1, d_2)$, which is perpendicular to the direction of greatest change. Directions are rounded to the nearest 45°.

If $d_m \geq t$, P is replaced by the average of its eight neighbors. Figure 6b shows four iterations for t = 2,3,4,5. Iterating smoothed the pictures quite well, but the pictures also became more blurred. The results are quite similar for all four t's.

A third variation uses four differences. In addition to the previously mentioned d_1 and d_2, the following are also calculated:

$$d_3 = |[g(i-1,j+1)+g(i,j+1)+g(i-1,j)] - [g(i+1,j)+g(i,j-1)+g(i+1,j-1)]|/3$$

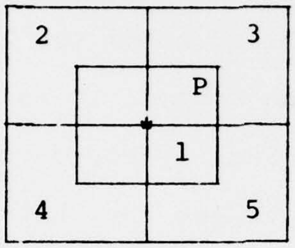
and

$$d_4 = |[g(i,j+1)+g(i+1,j+1)+g(i+1,j)] - [g(i-1,j)+g(i-1,j-1)+g(i,j-1)]|/3$$

The difference range in the neighborhood of P is calculated to be the absolute difference between the minimum and maximum of these four absolute differences. If this range exceeds a given threshold, t, P is replaced by the average of the two neighbors in the direction of the minimum absolute difference. Otherwise, P is replaced by the average of its eight neighbors. Four iterations for t = 2,3,4,5 are shown in Figure 6c. Results are better than those of the second variation, and are rather similar for all four t's. In the LANDSAT image, the edges become somewhat jagged or "furry".

2.6. Maximum homogeneity smoothing [7]

The gray level at a point P is replaced by the average gray level of the most homogeneous neighborhood from among the five 4x4 square neighborhoods surrounding P, as illustrated below:



To determine the most homogeneous neighborhood, the noisy image g(m,n) is first blurred by averaging over an nxn neighborhood of each point to obtain b(m,n). Non-homogeneity indexes are computed for five 2nx2n neighborhoods with centers at b(i,j), b(i+n,j+n), b(i-n,j+n), b(i-n,j-n) and b(i+n,j-n), using the formula

$$\begin{aligned} &\left|[b(p+\tfrac{n}{2},q+\tfrac{n}{2}) + b(p-\tfrac{n}{2},q+\tfrac{n}{2})] - [b(p+\tfrac{n}{2},q-\tfrac{n}{2}) + b(p-\tfrac{n}{2},q-\tfrac{n}{2})]\right| \\ &+ \left|[b(p+\tfrac{n}{2},q+\tfrac{n}{2}) + b(p+\tfrac{n}{2},q-\tfrac{n}{2})] - [b(p-\tfrac{n}{2},q+\tfrac{n}{2}) + b(p-\tfrac{n}{2},q-\tfrac{n}{2})]\right| \end{aligned}$$

where (p,q) represents the center of the neighborhood. In the examples, n = 2. The neighborhood for which this index is lowest is taken to be the most homogeneous neighborhood of P.

The results of four iterations are shown in Figure 7. Since the initial averaging weakens the noise, and averaging over the most homogeneous area further smooths the image, both pictures are very clean. However, edges are jagged and the LANDSAT image is very blocky. (The sharp appearance of the right and bottom edges of the third and fourth iterations is not due to the method; it is the result of replacing undefined edge points by corresponding points in the uncorrupted picture.) To see why, note that, for each edge point on the octagon, there will be at least one area which falls completely within the background and one which falls completely within the object. On a diagonal edge, areas 2 and 5 or 3 and 4 will have small non-homogeneity indexes. Thus the area over which the average is taken will vary, causing roughness in the edges.

In [7], the authors caution that this method assumes smooth boundaries between regions so that complexly-shaped boundaries may give erroneous results. This seems to be true in the LANDSAT image. Small features have disappeared; the narrow spaces between the larger features are also gone. The results also appear very blocky. Because any one neighborhood will be likely to cover more than one type of region, due to the amount of detail in the image, the most homogeneous region will be "random," at least the first time (see the first iteration). After this picture is blurred, the regions within the areas should be a little more homogeneous than before. Additional iterations cause "false" regions to be built.

2.7. Neighbor-weighting [8]

In this method, the gray level at a point P is replaced by a weighted average of its neighbors in an nxn neighborhood of P. Two types of weights were used: line-sensitive weights and contrast-sensitive weights.

Consider the neighborhood A of P. An edge or line can pass through A in twelve simple ways (see Figure 8). Corresponding to these twelve ideal A_i's, $1 \le i \le 12$, are twelve weight matrices, D_i, shown in Figure 9. Rarely will A match any of the A_i's exactly. More often, A will be a combination of several A_i's. If, in addition, A is noisy, then even if $A = A_i$, the corresponding D_i may not be the correct one, since the match may be due to the noise. Instead, coefficients w_i, which measure how closely A matches A_i, should be calculated so that the weight matrix has the form

$$D = \sum_{i=1}^{12} w_i D_i$$

The w_i's are calculated as follows: let S be the sum of all nine gray levels in A, let S_j be the sum of the three gray levels of the points in A which have the same positions as the a's in A_j, and let $O_j = 3S_j - S$. If A is one of the A_i's, we have

$$O_j(A_j) = 6(a-b)$$

$$O_j(A_i) = 3(a-b) \text{ if } A_i \text{ and } A_j \text{ are both edges and have adjacent slopes}$$

$$O_j(A_i) \le 0, \text{ otherwise (assuming } a>b)$$

This means that $O_j(A)$ is large if A closely resembles A_j. The w_i's can then be defined by

$$w_i = O_i+/\Sigma O_j^+$$

where $O_j^+ = O_j$ if $O_j \geq 0$, and 0 if $O_j < 0$. Only those $O_i(A_j)$'s for which A_j resembles A_i influence w_i. If all the O_i's have equal magnitude, the new point is just the average of all the nine points in the neighborhood. Note that the values of the w_i's are not affected by the gray levels of the points in the neighborhood.

The contrast-sensitive weights are based on evaluating the difference (or similarity) between a point and its immediate neighborhood. This similarity measure can detect the existence of local edges and lines and is sensitive to the gray level differences (contrasts) involved.

Let the neighborhood A contain the gray levels

abc
def
ghi

where e is the gray level of P. For direction 0°, define

$$D_0=\begin{pmatrix}\alpha & \alpha & \alpha \\ 1 & 1 & 1 \\ \beta & \beta & \beta\end{pmatrix}$$

where $\alpha = \exp(-|(a+b+c)-(d+e+f)|/\sigma)$; $\beta = \exp(-|(g+h+i)-(d+e+f)|/\sigma)$. For the examples in Figure 10, $\sigma = 5$. Similarly, D_i's are defined for directions 45°, 90° and 135°:

$$D_1 = \begin{pmatrix} \gamma & \gamma & 1 \\ \gamma & 1 & \delta \\ 1 & \delta & \delta \end{pmatrix} \qquad D_2 = \begin{pmatrix} \varepsilon & 1 & \zeta \\ \varepsilon & 1 & \zeta \\ \varepsilon & 1 & \zeta \end{pmatrix} \quad \text{and } D_4 = \begin{pmatrix} 1 & \eta & \eta \\ \theta & 1 & \eta \\ \theta & \theta & 1 \end{pmatrix}$$

where

$$\gamma = \exp(-|(a+b+d)-(c+e+g)|/\sigma),$$

$$\delta = \exp(-|(f+h+i)-(c+e+g)|/\sigma),$$

$$\varepsilon = \exp(-|(a+d+g)-(b+e+h)|/\sigma),$$

$$\zeta = \exp(-|(c+f+i)-(b+e+h)|/\sigma),$$

$$\eta = \exp(-|(b+c+f)-(a+e+i)|/\sigma),$$

$$\theta = \exp(-|(d+g+h)-(a+e+i)|/\sigma).$$

The product of the D_i, taken elementwise, is

$$\begin{array}{ccc} \alpha\gamma\varepsilon & \alpha\gamma\eta & \alpha\zeta\eta \\ \gamma\varepsilon\theta & 1 & \delta\zeta\eta \\ \beta\varepsilon\theta & \beta\delta\theta & \beta\delta\zeta \end{array}$$

The final weight matrix D is obtained by normalizing this product to make the center term 1/9, and to make the sum of all terms equal to 1. This product rule was chosen because any coefficient set to a very small value will cause that point to contribute very little to the next estimate for P. This corresponds to the behavior of one of the D_i in the presence of a very strong edge or line. It also prevents the additive accumulation of the small responses from the rotated D_i which occur in noisy situations.

Results of applying four iterations of the first method are shown in Figure 10a and corresponding results for the second method in Figure 10b. Method 1 seems to perform better than method 2 on both pictures; the LANDSAT picture acquires a blotchy appearance under method 2.

2.8. Weighted averaging [9]

A blurred image, b(m,n) is computed by averaging g(m,n) over a 3x3 neighborhood. Next, a difference image d(m,n) = g(m,n)-b(m,n) is computed. Both b(m,n) and d(m,n) are then divided into corresponding sections of 4x4 pixels. To achieve overlapping of sections which the author of [9] has found to give more pleasing images, the 4x4 sections of d(m,n) are expanded to include two additional layers of pixels on each side yielding overlapping 8x8 square sections.

A variance $\sigma_{D_k}^2$ is computed for each 8x8 section of d(m,n). The noise variances are $\sigma_n^2 = 25$ for the octagon and $\sigma_n^2 = 64$ for the LANDSAT image. For each 4x4 section θ_k is defined by

$$\theta_k = \text{minimum}\{1.0,\ \sigma_n^2/\sigma_{D_k}^2\}$$

A new image $\hat{f}(m,n)$ is computed in 4x4 sections, where each point of the kth section is found by

$$\hat{f}(i,j) = \theta_k b(i,j)+(1-\theta_k)g(i,j)$$

When tested against σ_n^2, $\sigma_{D_k}^2$ is a measure of the signal activity in the kth section of the picture. If $\sigma_{D_k}^2 \le \sigma_n^2$, the original picture is assumed to have low signal activity and the kth section of the blurred image is assumed to be an adequate representation of the original. If $\sigma_{D_k}^2 > \sigma_n^2$, the signal is assumed to have high activity and a weighted average, as defined above, is computed for the kth section. (A related method is proposed in [10]; however, it is more a technique for detection of small, low-contrast objects in an image than for image smoothing.)

The blurred and smoothed images are shown in Figure 11. Little difference can be seen between the two.

2.9. Kalman filtering [1]

This method is included as a basic example of a sequential method. Several refinements of the Kalman filtering technique exist (e.g., [10,11]), but this version was implemented because of its simplicity. The three immediate neighbors on the left and above, as well as the point itself, are used in the estimation of the new value of the point:

$$\hat{f}(m,n) = \sum_{(i,j)\in D} d_{m,n}(i,j)\hat{f}(m-i,n+j)+\eta_{m,n}g(m,n)$$

where $D = \{(0,1),(1,1),(1,0)\}$. For each (m,n), $d_{m,n}(i,j)$, $(i,j)\in D$ and $\eta_{m,n}$ are the four coefficients of the filter. The method used in computing these coefficients is described in the following paragraphs.

The autocorrelation function of the ensemble to which the image belongs is assumed to be of the form

$$R_{ff}(\alpha,\beta) = R_{ff}(0,0)\rho_h^{|\alpha|}\rho_v^{|\beta|}$$

where $\rho_h = e^{-c_1}$ and $\rho_v = e^{-c_2}$ are measures of the horizontal and vertical correlations respectively. ρ_h and ρ_v are approximated by first calculating

$$R_{ff}(\alpha,\beta) = \sum_{k=1}^{m-\alpha}\sum_{\ell=\beta+1}^{n} g(k,\ell)g(k+\alpha,\ell-\beta)$$

for $\alpha = 0$, $\beta = 1,2,3$ and $\alpha = 1,2,3$, $\beta = 0$. After dividing the results by $R_{ff}(0,0)$, a least squares adjustment is done to obtain ρ_h and ρ_v.

Normally, a large amount of computation would be required to determine $\eta_{m,n}$. However, this quantity has been found to attain practically constant value only a few points away from the edges of a picture. This implementation assumes that edge points are noise free and, rather than calculating $\eta_{m,n}$, various constant values η between 0 and .6 are used.

The remaining coefficients are calculated using

$$d_{m,n}(i,j) = C_{i,j}(1-\eta), \quad (i,j) \in D$$

where $C_{1,0} = \rho_h$, $C_{0,1} = \rho_v$ and $C_{1,1} = -\rho_h \rho_v$.

As mentioned in [11], the mean of the image is subtracted from each point. All operations are performed on this zero mean picture; the mean is added back at the end. This method also has a directional bias, so filtering is first done from top to bottom, left to right, then from bottom to top, right to left. The results are averaged and this averaged value becomes the value at P.

Results for η = .1,.2,.3,.4,.5,.6 are shown in Figure 12. The values of ρ_h and ρ_v were .9754 and .9754 for the octagon and .9173 and .9269 for the LANDSAT image. The pictures have a textured appearance, as if they were printed on cloth. The images seem to be smoother, but more blurred, for smaller η's.

3. Comparative discussion and conclusions

Although general conclusions cannot be drawn on the basis of so few examples, a few tentative observations can be made.

The following methods were judged to perform more poorly than the others:

1) Mode filtering (Section 2.1) produced strong mottle
2) Maximum homogeneity (Section 2.6) produced blocky output
3) Neighbor-weighting, second method (Section 2.7) made the LANDSAT image blotchy
4) Kalman filtering (Section 2.9) produced a periodic "cheesecloth" pattern

Note that the last two of these methods were not iterated.

Among the remaining methods, the best results on the LANDSAT image seem to be those obtained by median filtering (Section 2.1), gradient smoothing (Section 2.4), and the first neighbor-weighting method; the last of these also yielded the least mottled octagon. Among the E^k methods (Section 2.3), E^4 was somewhat too noisy and E^6 somewhat too blurry. For the selective averaging methods (Section 2.5), the first yielded noisier results than the other methods mentioned in this paragraph; the third yielded good results (and less blurry than those of the second), but both introduced "furriness" into the edges. In all of the selective averaging methods, the choice of the threshold t, over the ranges considered, did not appear to be highly significant.

Although iteration tended to improve most of the results, the point at which blurring effects became dominant could not be determined beforehand. In the case of some of the methods, specifically, E^k and neighbor weighting, considerable computer time would be saved if this information were available.

The evaluation of these results was very subjective. One possible way of obtaining a more effective evaluation might be to ask a number of people to evaluate the results according to a set of written instructions. These results could then be tabulated and analyzed. In addition, these methods could be applied to a wider variety of pictures. The performance of more advanced methods such as [11-16] could also be studied.

References

1. Rosenfeld, A. and A. Kak. Digital Picture Processing, Academic Press, N.Y., 1976.

2. Rosenfeld, A. Iterative Methods in Image Analysis, Univ. of Maryland Computer Science Center TR-517, April 1977.

3. Davis, L.S. and A. Rosenfeld. Noise Cleaning by Iterated Local Averaging, Univ. of Maryland Computer Science Center TR-520, April 1977.

4. Pratt, W. K. Private communication.

5. Pratt, W. K. Median Filtering, in Semiannual Report, Image Processing Institute, Univ. of Southern California, Sept. 1975, pp. 116-123.

6. Final Report, Algorithms and Hardware Technology for Image Recognition, Computer Science Center, Univ. of Maryland, March 1978.

7. Tomita, F. and S. Tsuji. Extraction of Multiple Regions by Smoothing in Selective Neighborhoods, IEEE Transactions on Systems, Man and Cybernetics 7, 1977, 107-109.

8. Lev, A., S. W. Zucker and A. Rosenfeld. Iterative Enhancement of Noisy Images, ibid., 435-442.

9. Trussell, H. J. A Fast Algorithm for Noise Smoothing Based on a Subjective Criterion, ibid., 677-678.

10. Zweig, H. J., E. B. Barrett and P. C. Hu. Noise-cheating Image Enhancement, Journal of the Optical Society of America 65, 1347-1353.

11. Panda, D. and A. Kak. Recursive Least Squares Smoothing of Noisy Images, IEEE Transactions on Acoustics, Speech and Signal Processing, ASSP-25, 1977, 520-524.

12. Nahi, N. E. and A. Habibi. Decision-Directed Recursive Image Enhancement, IEEE Transactions on Circuits and Systems CAS-22, 1975, 286-293.

13. Woods, J. W. and C. H. Radewan. Kalman Filtering in Two Dimensions, IEEE Transactions on Information Theory 23, 1977, 473-482.

14. Jain, A. K. A Semicausal Model for Recursive Filtering of Two-Dimensional Images, IEEE Transactions on Computers 26, 1977, 343-350.

15. Keshavan, H. R. and M. D. Srinath. Sequential Estimation Technique for Enhancement of Noisy Images, ibid., 971-987.

16. Keshavan, H. R. and M. D. Srinath. Interpolative Models in Restoration and Enhancement of Noisy Images, IEEE Transactions in Acoustics, Speech and Signal Processing ASSP-25, 1977, 525-534.

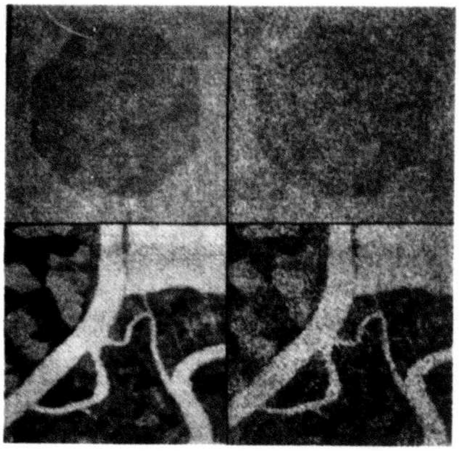


Figure 1

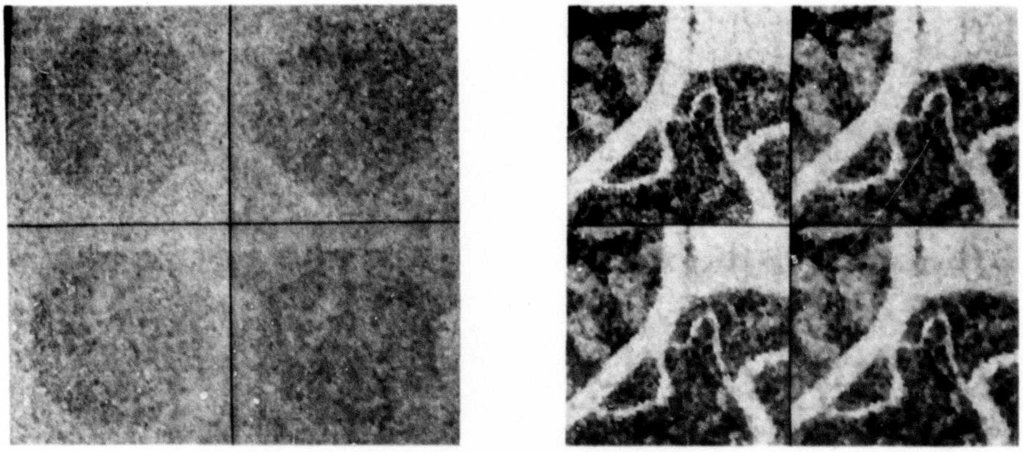


Figure 2. Mode Filtering

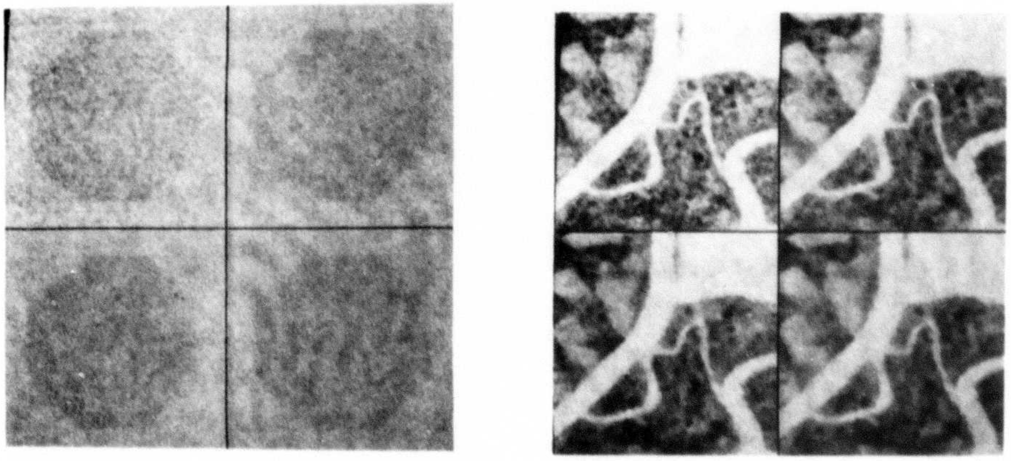
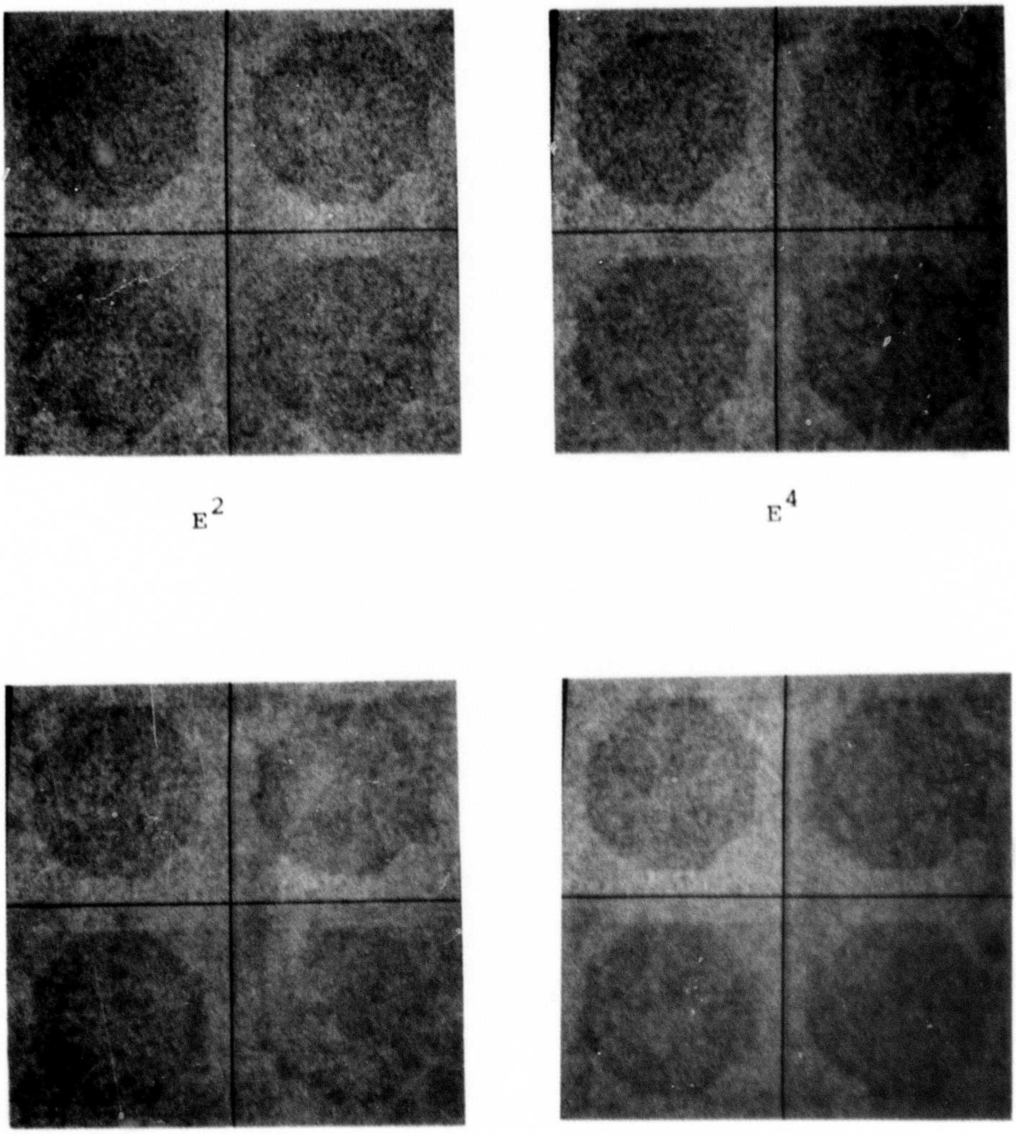


Figure 3. Median Filtering



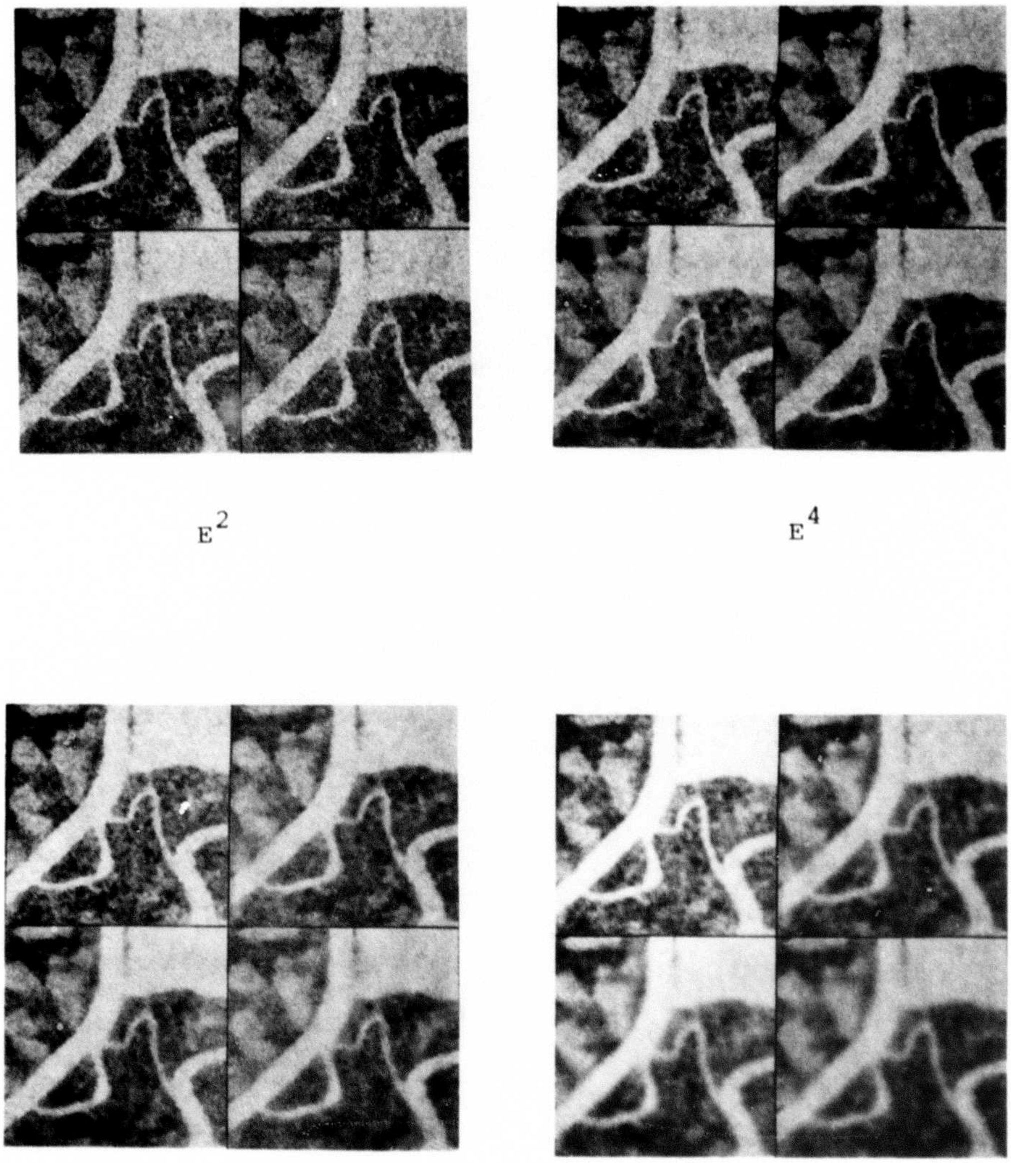
E^2

E^4

E^6

E^8

Figure 4. E^k



E^2

E^4

E^6

E^8

Figure 4. continued

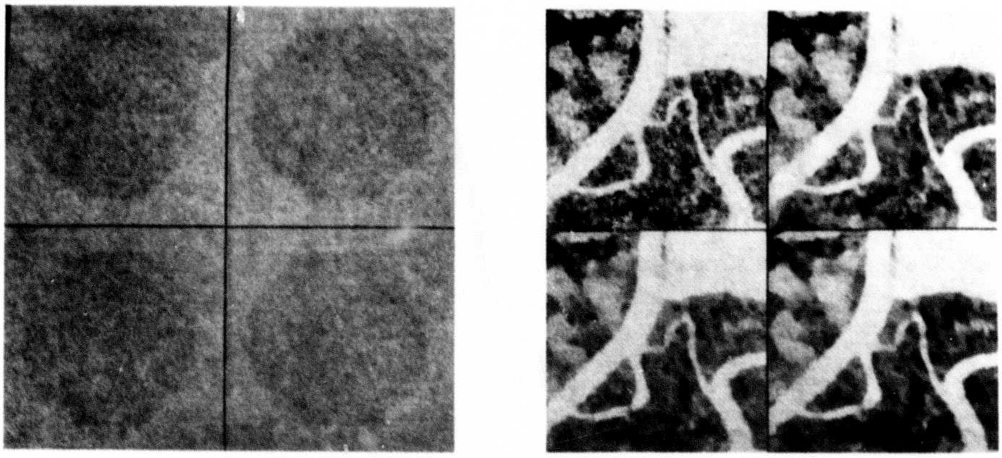
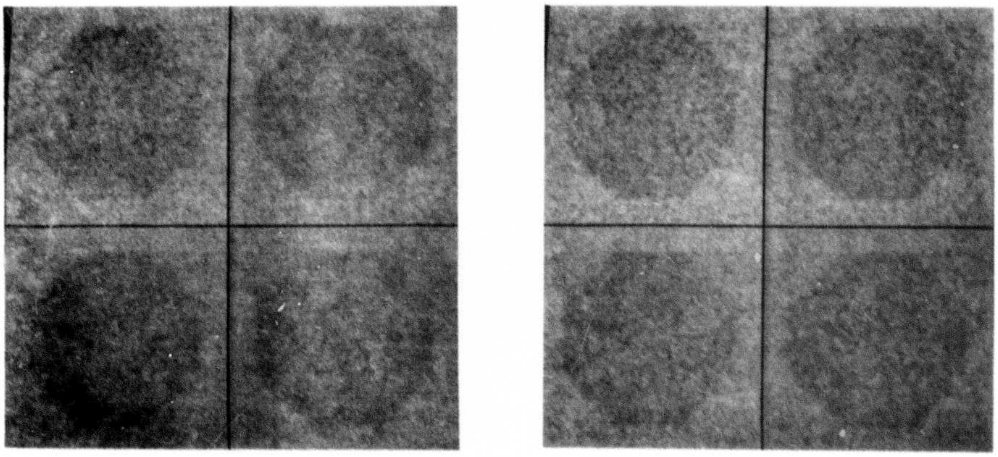


Figure 5. Gradient Smoothing



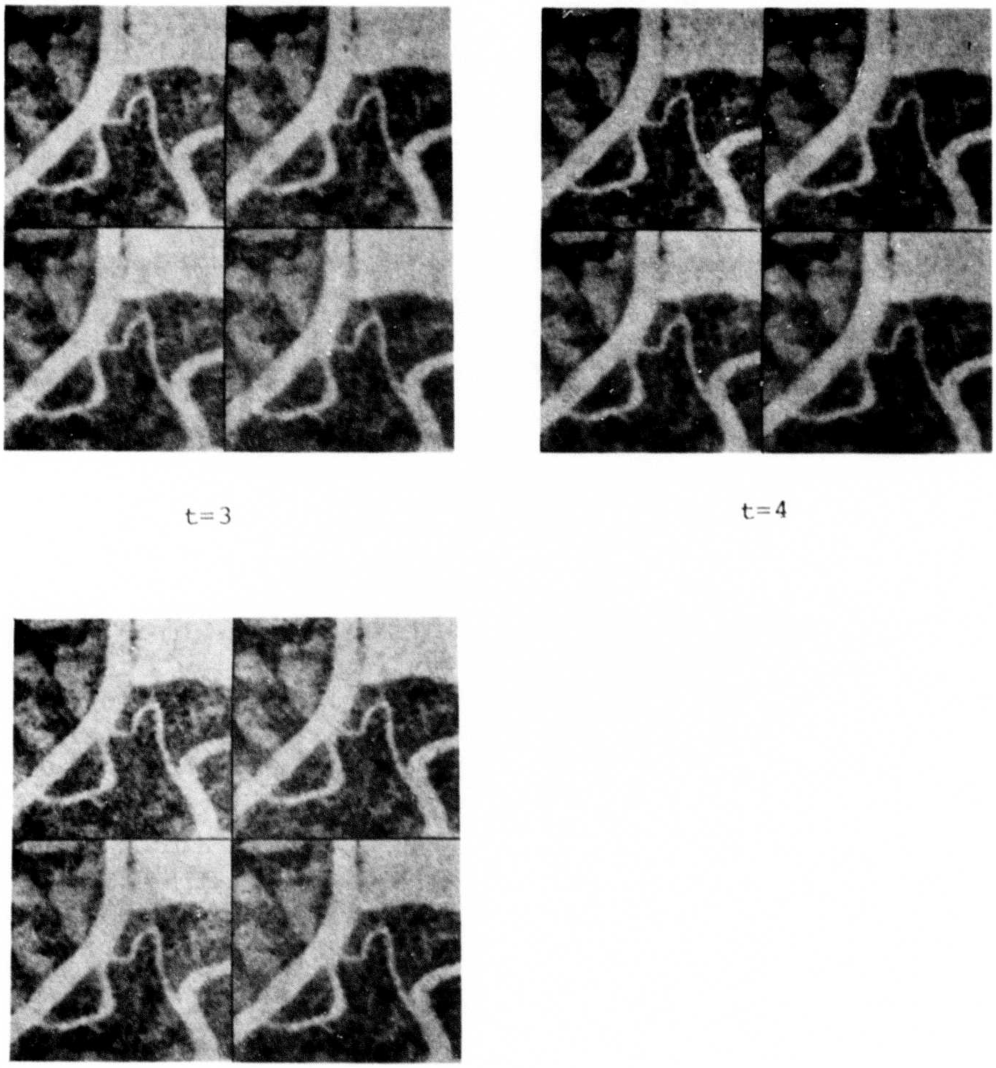
t=2

t=3

(a)

Figure 6. Selective Averaging

a) Variation 1 b) Variation 2 c) Variation 3

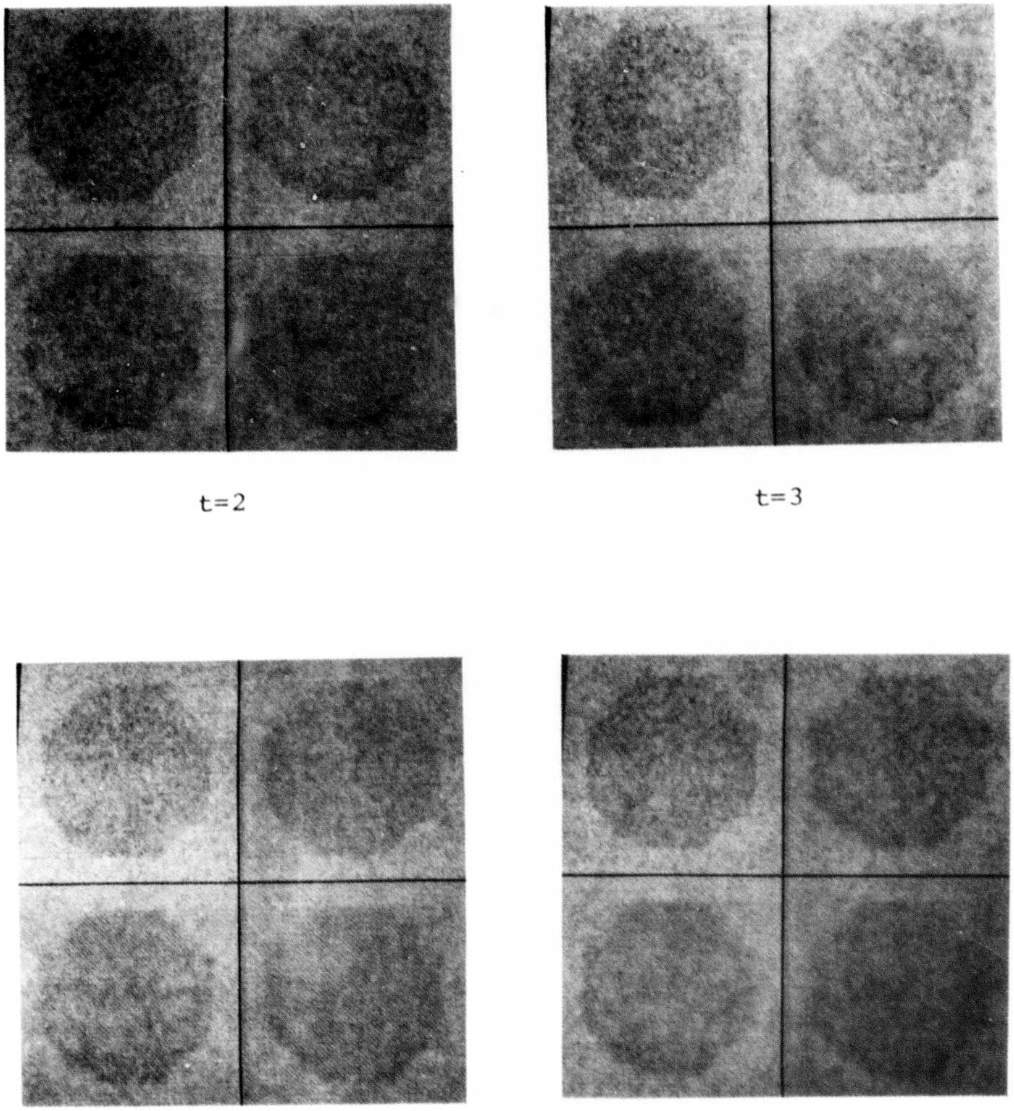


t=3

t=4

t=5

Figure 6a. continued



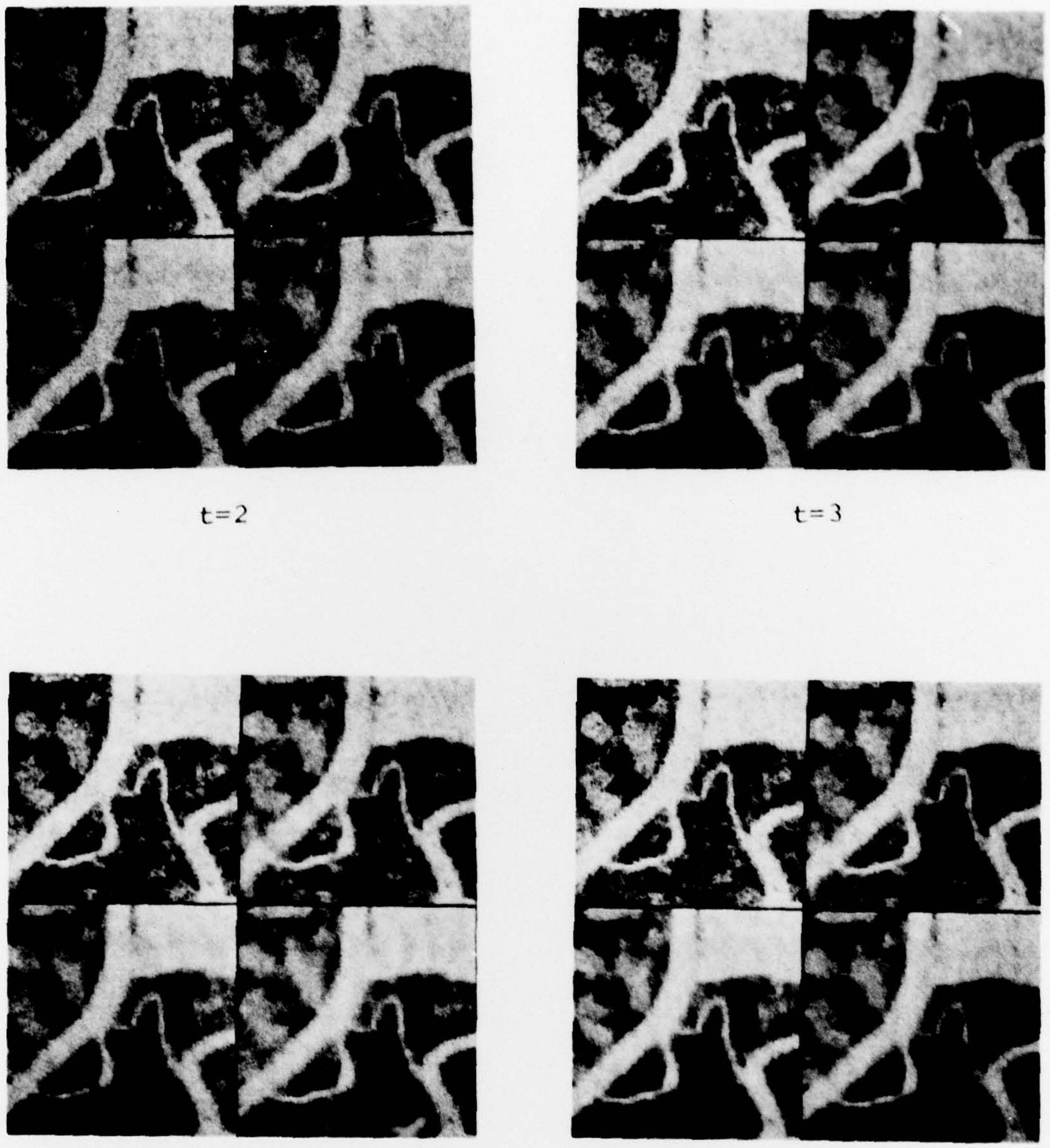
t=2

t=3

t=4

t=5

Figure 6b.



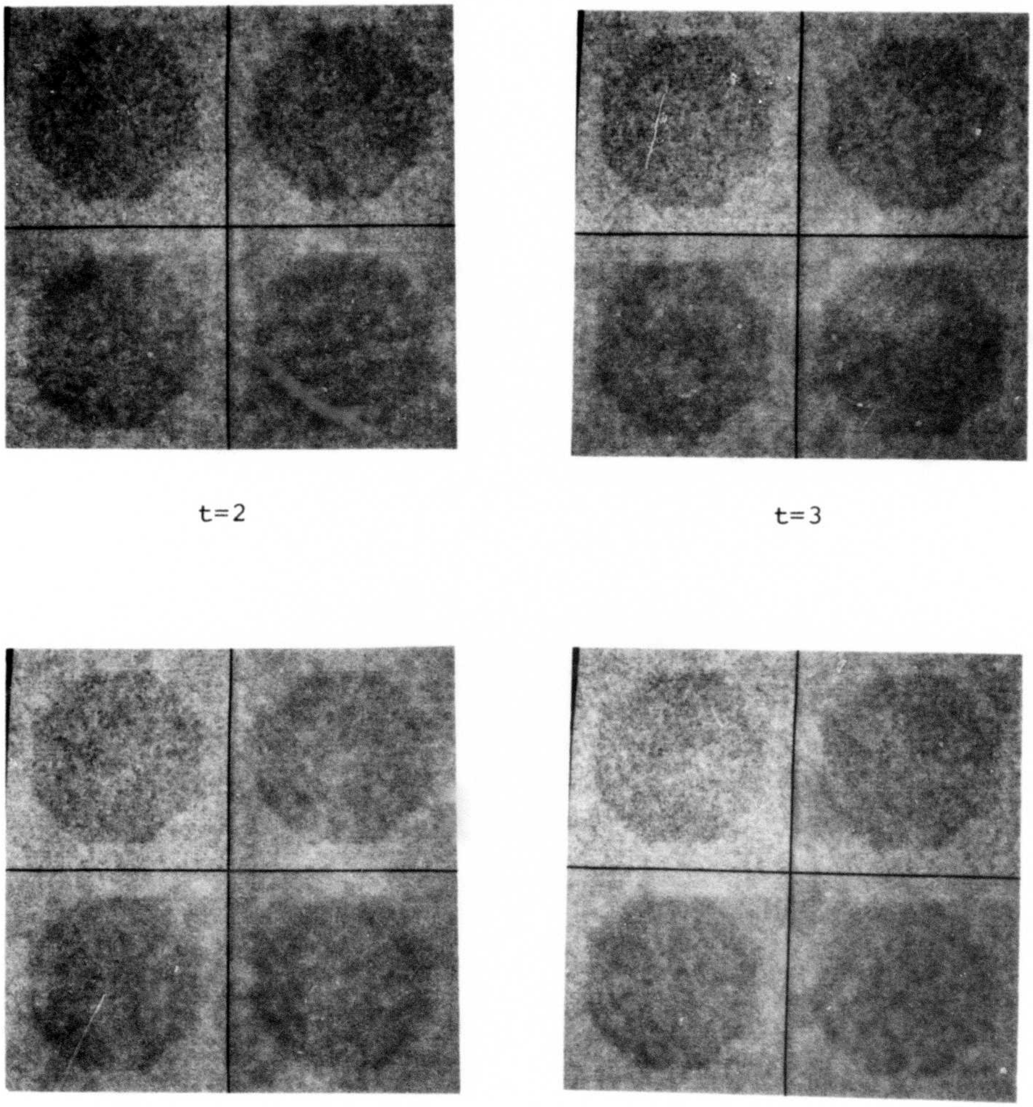
t=2

t=3

t=4

t=5

Figure 6b. continued



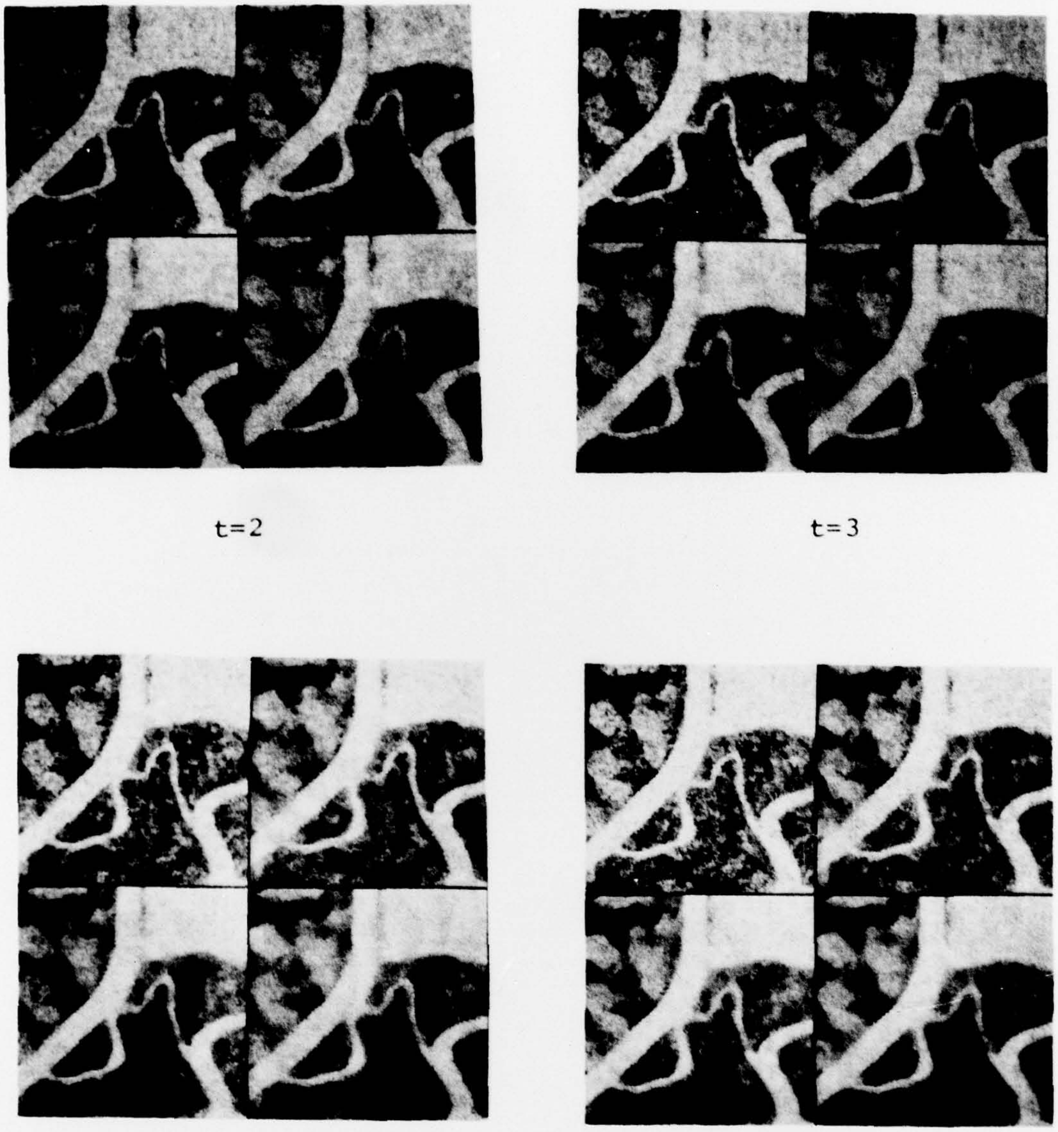
t=2

t=3

t=4

t=5

Figure 6c



t=2

t=3

t=4

t=5

Figure 6c. continued

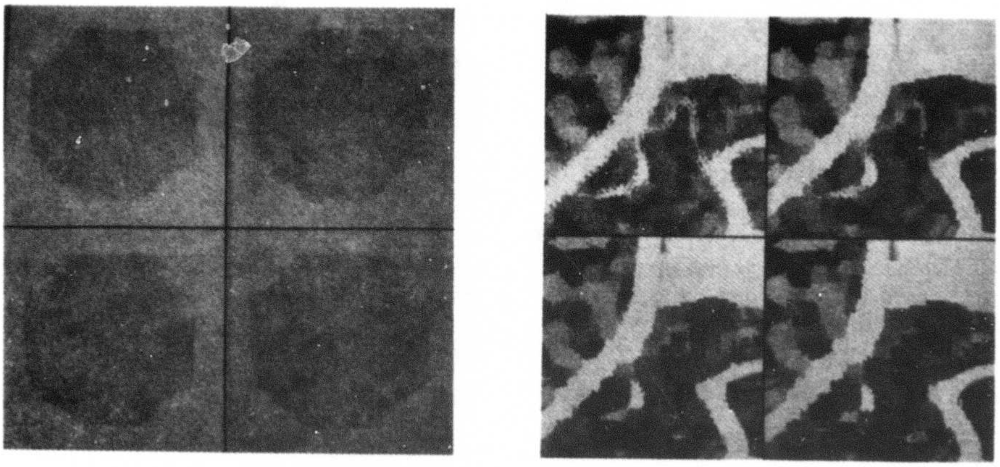


Figure 7. Maximum Homogeneity Smoothing

$$A_1 = \begin{matrix} a & a & a \\ b & b & b \\ b & b & b \end{matrix} \qquad A_2 = \begin{matrix} a & a & b \\ a & b & b \\ b & b & b \end{matrix} \qquad A_3 = \begin{matrix} a & b & b \\ a & b & b \\ a & b & b \end{matrix} \qquad A_4 = \begin{matrix} b & b & b \\ a & b & b \\ a & a & b \end{matrix}$$

$$A_5 = \begin{matrix} b & b & b \\ b & b & b \\ a & a & a \end{matrix} \qquad A_6 = \begin{matrix} b & b & b \\ b & b & a \\ b & a & a \end{matrix} \qquad A_7 = \begin{matrix} b & b & a \\ b & b & a \\ b & b & a \end{matrix} \qquad A_8 = \begin{matrix} b & a & a \\ b & b & a \\ b & b & b \end{matrix}$$

$$A_9 = \begin{matrix} b & b & b \\ a & a & a \\ b & b & b \end{matrix} \qquad A_{10} = \begin{matrix} b & b & a \\ b & a & b \\ a & b & b \end{matrix} \qquad A_{11} = \begin{matrix} b & a & b \\ b & a & b \\ b & a & b \end{matrix} \qquad A_{12} = \begin{matrix} a & b & b \\ b & a & b \\ b & b & a \end{matrix}$$

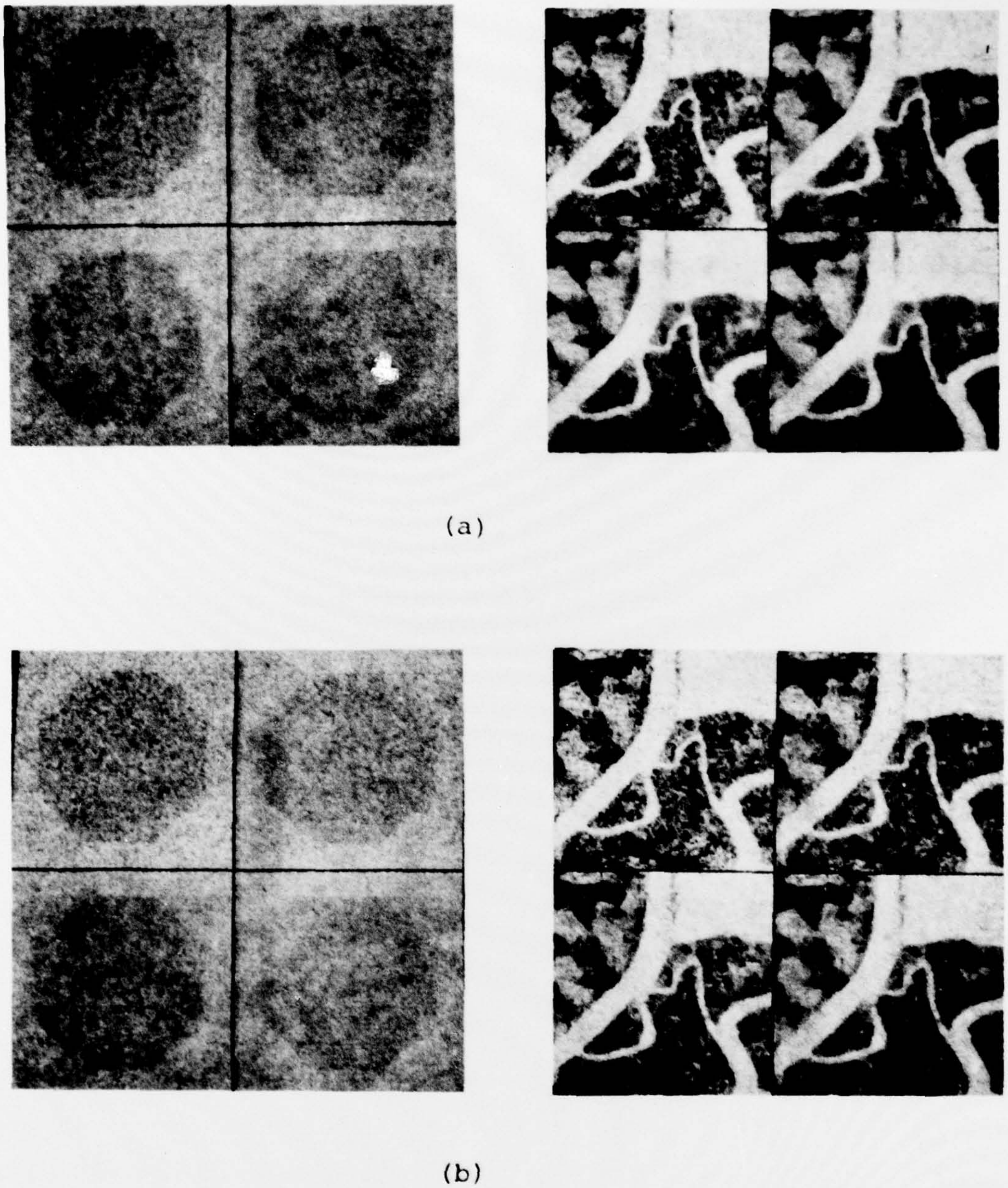
Figure 8: Twelve ways in which an edge or line can pass through the neighborhood of P.

$$D_1 = \frac{1}{6}\begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \qquad D_2 = \frac{1}{6}\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \qquad D_3 = \frac{1}{6}\begin{pmatrix} 0 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} \qquad D_4 = \frac{1}{6}\begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$$D_5 = \frac{1}{6}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix} \qquad D_6 = \frac{1}{6}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \qquad D_7 = \frac{1}{6}\begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{pmatrix} \qquad D_8 = \frac{1}{6}\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

$$D_9 = \frac{1}{6}\begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix} \qquad D_{10} = \frac{1}{6}\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \qquad D_{11} = \frac{1}{6}\begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix} \qquad D_{12} = \frac{1}{6}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Figure 9: Weight matrices corresponding to the twelve ideal neighborhoods of Figure 8.



(a)

(b)

Figure 10. Neighbor-Weighting

a) Method 1 b) Method 2

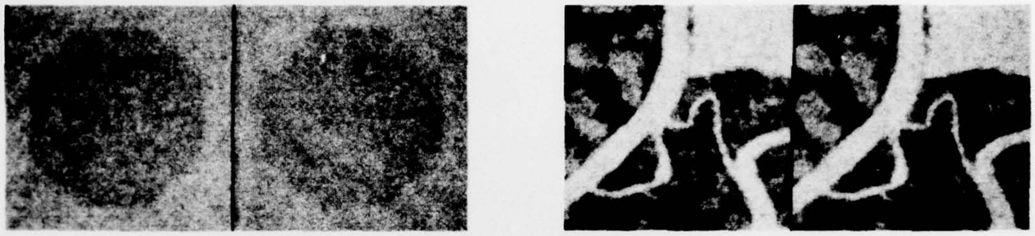


Figure 11. Weighted Averaging

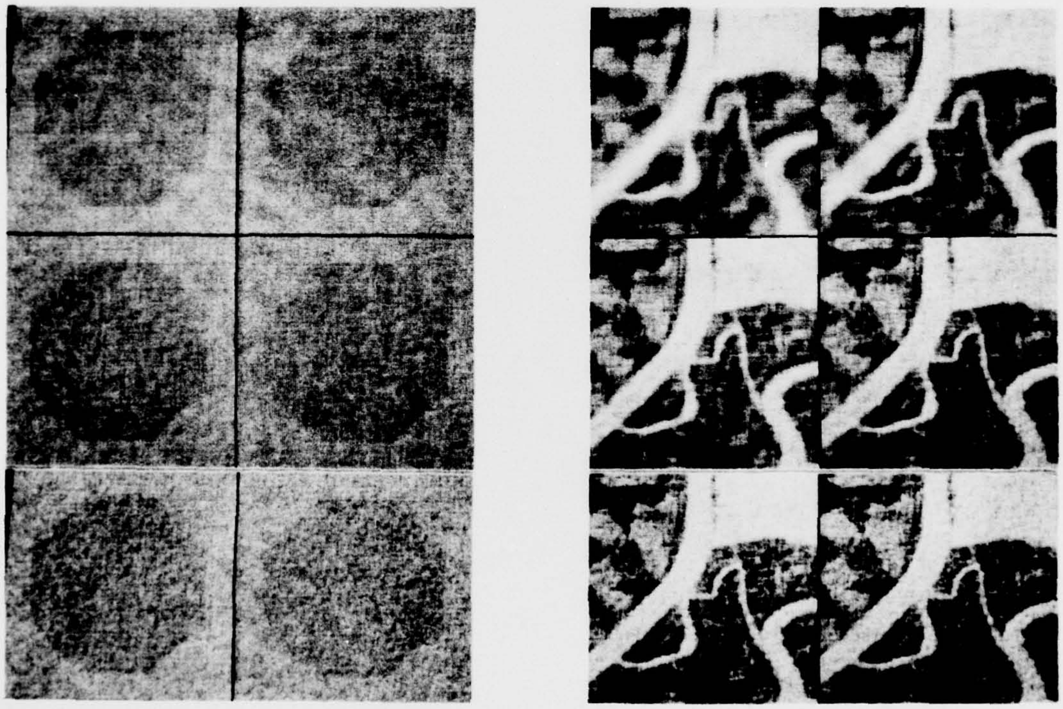


Figure 12. Kalman Filtering η = 0.1 - 0.6